A Theme-based Project Report
On

AIR POLLUTION MONITORING SYSTEM

Submitted in partial fulfilment of the
Requirements for the award of the Degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE & ENGINEERING

By

K. GOPI KRISHNA

1602-19-733-069

B. SRIDEEP

1602-19-733-109

Department of Computer Science & Engineering

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Ibrahimbagh, Hyderabad-31

2022

ACKNOWLEDGEMENT

I would like to express our heartfelt gratitude to S. Vinay Kumar sir, our project guide, for his valuable guidance and constant support, along with his capable instruction and persistent encouragement.

I am grateful to our Head of Department, Dr. T. Adilakshmi, for her steady support and the provision of every resource required for the completion of this project.

I would like to take this opportunity to thank our Principal, Dr. S. V. RAMANA, as well as the management of the institute, for having designed an excellent learning atmosphere.

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Air pollution affects our day-to-day activities and quality of life. It poses a threat to the ecosystem and the quality of life on the planet. The dire need to monitor air quality is very glaring, owing to increased industrial activities over the past years. People need to know the extent to which their activities affect air quality. This project proposes an air pollution monitoring system. The air pollution monitoring system was designed to monitor and analyse air quality in real-time and log data to a remote server, keeping the data updated over the internet. Air quality measurements were taken based on the Parts per Million (PPM) metrics.

INTRODUCTION

Air is one of the essential elements of man’s surroundings. The earth’s atmosphere is full of air which contains gases such as Nitrogen, Oxygen, Carbon Monoxide, and traces of some rare elements. Humans need an atmosphere of air that is free from contaminants. This is very crucial for human life and health. Any change in the natural composition of air may cause grave harm to life forms on earth. Air pollution is the presence of one or more contaminants in the atmosphere such as gases in a quantity that can harm humans, animals, and plants. Air pollutants are measured in Parts per Million (ppm) or ug/m3. Primary pollutants are released directly into the atmosphere. Secondary pollutants are produced when the primary pollutant reacts with other atmospheric chemicals. Air quality affects public health. The effect of air pollution ranges from difficulty in breathing, coughing, aggravation of asthma and emphysema . Polluted air can also impair visibility. Air pollution is accountable for the death of 7 million persons worldwide each year or one in eight premature deaths yearly . Almost 570,000 children under the age of five die every year from respiratory infection linked to indoor/outdoor pollution and second-hand smoke . Children exposed to air pollution have an elevated risk of developing chronic respiratory problems such as asthma. In the monitoring of air pollution, several researchers worldwide have developed models to monitor many of the pollution gases such as Sulphur Dioxide (SO2), Carbon Monoxide (CO), Carbon Dioxide (CO2), Nitrogen Oxides (NO) etc. In this project we Analyse particulate matter in the surrounding by interfacing SDS sensor with ESP8266. Understand the behaviour of particulate matter with respect to variation of temperature and humidity. Develop a real-time deployable IoT node which sends data to thingspeak.

SOFTWARE REQUIREMENT SPECIFICATION

The Internet of Things (IoT) is a sprawling set of technologies and use cases that has no clear, single definition. One workable view frames IoT as the use of network-connected devices, embedded in the physical environment, to improve some existing process or to enable a new scenario not previously possible.

The system uses Node MCU. These devices, or things, connect to the network to provide information they gather from the environment through sensors, or to allow other systems to reach out and act on the world through actuators.

Hardware Specifications

- ESP8266 - 1 pcs
- SDS 011 (for Pm2.5 and Pm10) - 1 pcs
- AHT10 (for Temperature and Humidity) – 1 pcs
- Jumper Wires
- Breadboard
- USB Cable - 1 pcs

Software Specifications

- Arduino IDE Compiler

WORKING

The air quality measurement sensor has a built-in air circulation fan. This fan helps to circulate the air through the inlet valve to the output valve of the dust sensor. Moreover, it circulates air with a laser diode which is used to determine the size of dust particles in the air. When air enters the dust sensor through an inlet valve (as shown on the left-hand side of the SDS011 internal circuit) a laser diode illuminates the air and produces a scattered light. In response, the photodiode detects this scattered light and converts it into electrical signals. After that, these electrical signals are amplified through an amplification circuit and processed to get an amount of Pmi’s in the air that is the number and diameter of dust particles.

ARCHITECTURAL DIAGRAM:

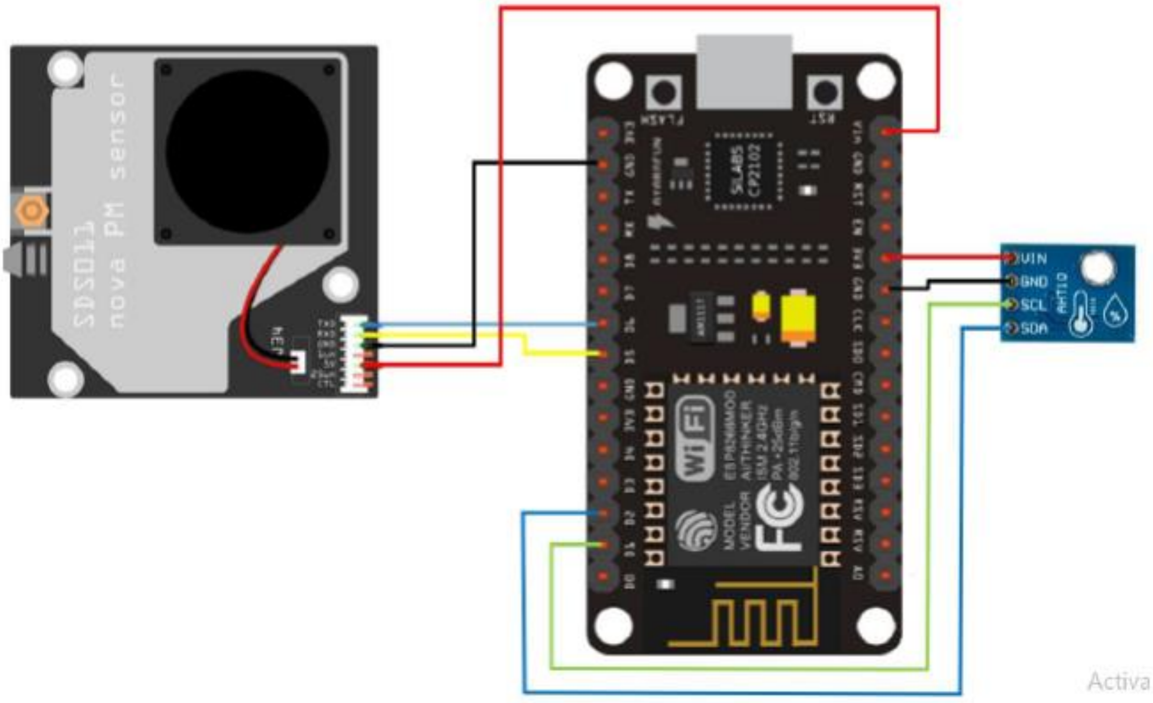


FIGURE-1: ARCHITECTURE

NODEMCU ESP8266

The NodeMCU ESP8266 development board comes with the ESP-12E module containing the ESP8266 chip having Tensilica Xtensa 32-bit LX106 RISC microprocessor. This microprocessor supports RTOS and operates at 80MHz to 160 MHz adjustable clock frequency. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with in-built Wi-Fi / Bluetooth and Deep Sleep Operating features make it ideal for IoT projects.

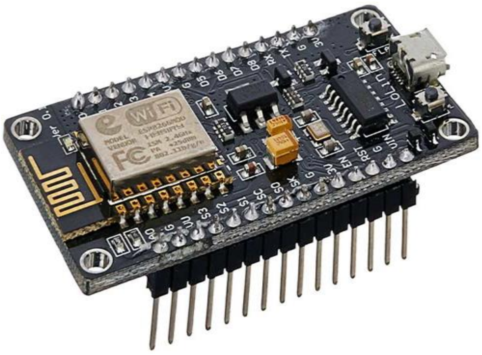


FIGURE-2: ESP8266

SDS 011

SDS 011 is an air quality measurement sensor which can be used to get dust particles and smoke concentration in the air. More precisely, it can measure particulate matter (PM) concentrations in the air. It can detect the dust particles concentration between 0.3 to 10um. Most importantly, Nova PM dust sensor provides an interrupt-based response when the concentration of dust particles changes in the air and the response time is less than 10 seconds. The operating voltage range is 4.7-5.3V which makes it suitable to use with standard voltage of 5 volts. It consists of 7 pins. But to interface it with microcontrollers such Arduino and ESP32, we can use only UART pins, such as RX, TX, to get dust concentration output from the dust sensor.

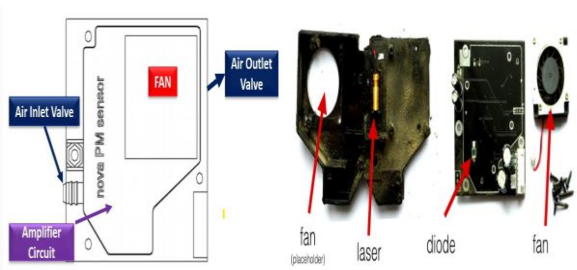


FIGURE-3: SDS 011

JUMPER WIRES

A jump wire (also known as jumper, jumper wire, DuPont wire) is an electrical wire, or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering.

FIGURE-4: JUMPER WIRES

AHT-10 SENSOR

AHT10 is a high precision, fully calibrated, temperature and humidity sensor chip package, the MEMS production process, to ensure that products with high reliability and excellent long-term stability. The sensor includes a capacitive humidity sensing element, and a high-performance microprocessor CMOS is connected. The product has excellent quality, fast response, anti-strong anti-interference ability, cost performance advantages.

FIGURE-5: AHT-10

BREAD BOARD:

A breadboard consists of plastic block holding a matrix of electrical sockets of a size suitable for gripping thin connecting wire, component wires or the pins of transistors and integrated circuits (ICs). The sockets are connected inside the board, usually in rows of five sockets.

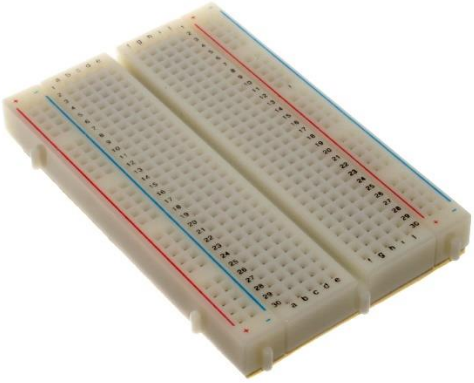


FIGURE-6: BREAD BOARD

IMPLEMENTATION

code:

```
#include <SDS011.h>

#include <Adafruit_AHTX0.h>

#include <ESP8266WiFi.h>

#include<ThingSpeak.h>

String apiKey = "9QYFK4UB8UPJMZRO";

const char* ssid = "Stark 4G";

const char* password = "12345666";

const char* server = "api.thingspeak.com";

float p10,p25;

int error;

SDS011 my_sds;

Adafruit_AHTX0 aht;

WiFiClient client;

void setup() {

  Serial.begin(9600);

   my_sds.begin(14,12);

 Serial.println("Adafruit AHT10/AHT20 demo!");

 if (! aht.begin()) {

 Serial.println("Could not find AHT? Check wiring");

 while (1) delay(10);

 }

  Serial.println("AHT10 or AHT20 found");
```

```
   WiFi.begin(ssid, password);

   while (WiFi.status() != WL_CONNECTED) {

    Serial.print(".");

  }

}

void loop() {

  error = my_sds.read(&p25,&p10);

  if (! error) {

    Serial.println("P2.5: "+String(p25));

    Serial.println("P10: "+String(p10));

  }

  delay(100);

  sensors_event_t humidity, temp;

  aht.getEvent(&humidity, &temp);// populate temp and humidity objects with
fresh data

  if (client.connect(server,80))  {   //--> "184.106.153.149" or
api.thingspeak.com


    String postStr = apiKey;

    postStr +="&field3=";

    postStr += String(p25);

    postStr +="&field4=";

    postStr += String(p10);

    postStr +="&field1=";
    postStr+=String(temp.temperature);
    postStr+="&field2=";
    postStr+= String(humidity.relative_humidity);
    postStr += "\r\n\r\n";

    client.print("POST /update HTTP/1.1\n");
    client.print("Host: api.thingspeak.com\n");
    client.print("Connection: close\n");
    client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
    client.print("Content-Type: application/x-www-form-urlencoded\n");
    client.print("Content-Length: ");
    client.print(postStr.length());
    client.print("\n\n");
    client.print(postStr);

    Serial.print("Temperature: ");
    Serial.print(temp.temperature);
    Serial.println(" degrees C");
    Serial.print("Humidity: ");
    Serial.print(humidity.relative_humidity);
    Serial.println("% rH");
  }
  delay(500);
}
```

RESULTS

Snapshot of The Output

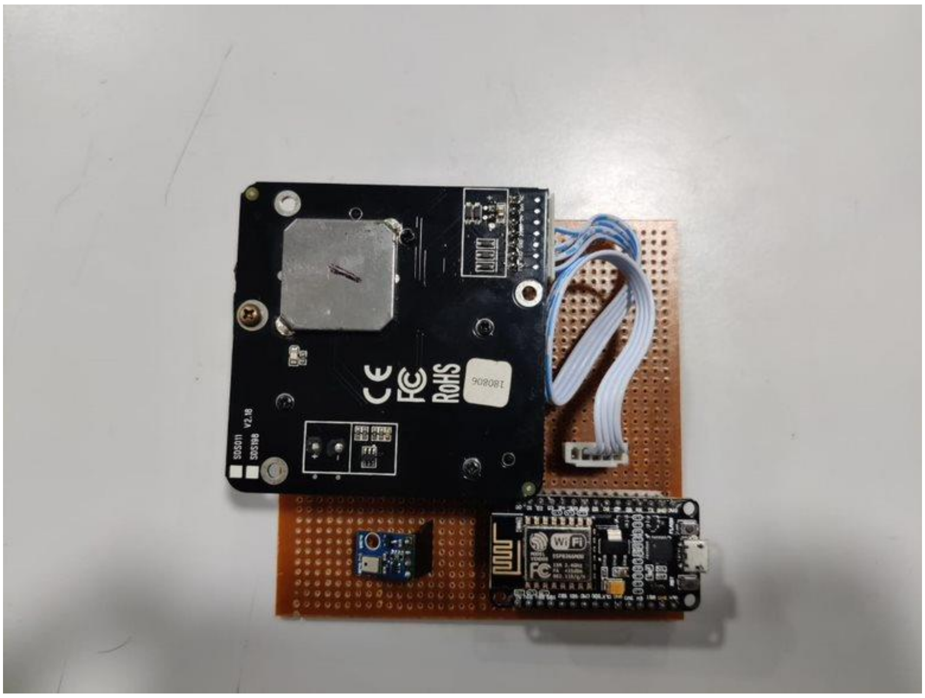
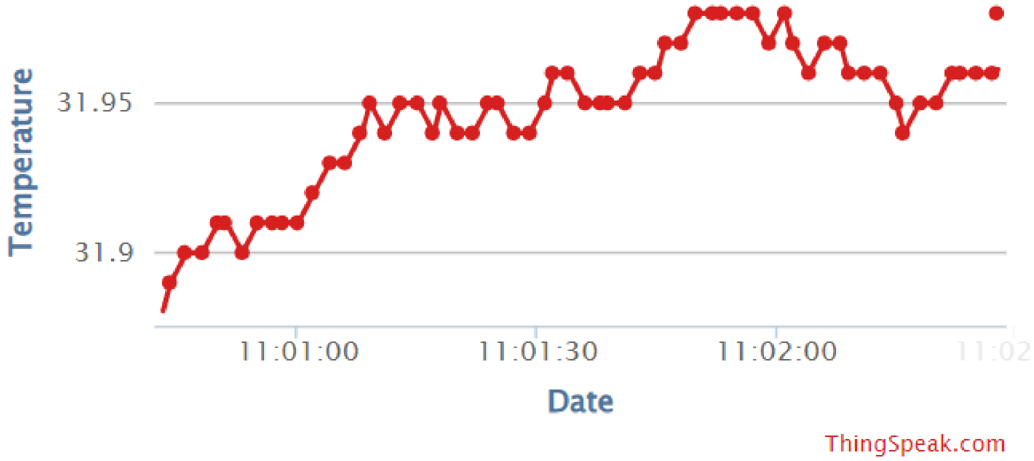
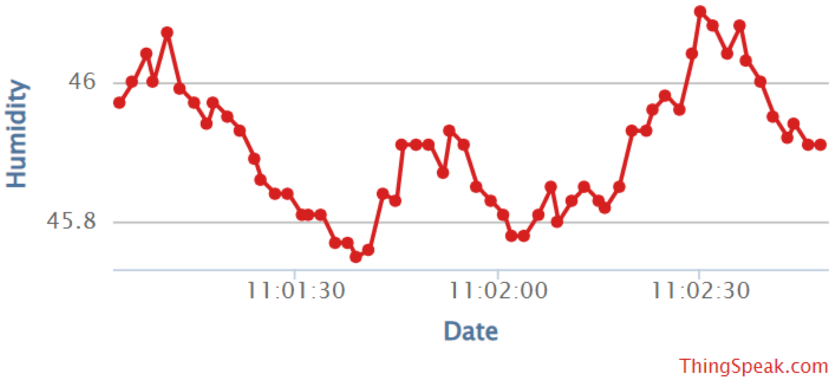


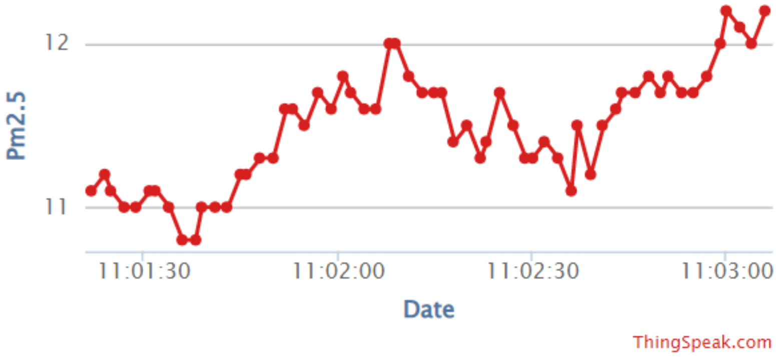
FIGURE-7: HARDWARE SETUP



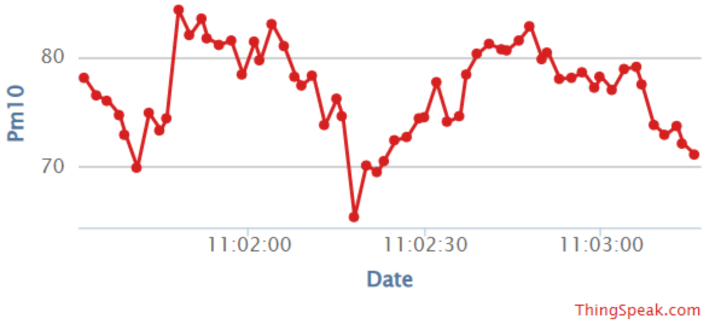
TEMPERATURE MONITORING IN THINGSPEAK



HUMIDITY MONITORING IN THINGSPEAK



PM2.5 MONITORING IN THINGSPEAK



PM10 MONITORING IN THINGSPEAK

CONCLUSION & FUTURE WORK

As a result, our project is to check the quality of the exposed level in the air pollution. The mobile application is developed by getting the source and destination address from the user. In this application it monitors the pollutant level through that way. It is also tracking the individual’s exposure level of air pollutants for a single day. Our project was designed to help a person to detect, monitor, and test air pollution in a given area. The kit has been integrated with a mobile application that helps the user in predicting the pollution level of their entire route. This proposed air pollution monitoring kit along with the integrated mobile application can be helpful to people to identify their exposure level to air pollutants. The app had following features, indices of air quality using real-time computation, air quality daily reports based on users travel distance, specific reports for air quality measures based on locations. Air Pollution is the major affecting factor to our environment. Not only affecting the environment and also affects the human health. The mobile application is developed to monitoring system it tracking the how much the human has exposed in a day. The SDS 011 sensor senses the gases and convert from analog to digital and displays in the application. The exposed level is calculated in PPM (Parts per Million).

REFERENCES

[1] IoT Based Air Pollution Monitoring System, Anand Jayakumar Arumugham, Suguna College of Engineering.

[2] A Smart Air Pollution Monitoring System, Kennedy Okokpujie, Covenant University Ota Ogun State, Nigeria.

[3] https://circuitdigest.com/microcontroller-projects/air-quality-analyzer-using-arduino-and-nova-air-quality-sensor-sds011

[4] https://circuitdigest.com/microcontroller-projects/aht-10

[5] The Changing Paradigm of Air Pollution Monitoring, Emily G. Snyder, Timothy H. Watkins.